High RISC, High Reward: The Journey from Ratified Specs to RVA23 Superclusters

Tom Gall
November 20th, 2026

A talk in three parts

0 - RISC-V International

1 - RISC-V specifications

2 - RVA23 - Clusters for HPC/AI - Are we there yet?



RISC-V is an ISA


RISC-V is an Open International Standard
ISO standard - RSN


RISC-V International is
headquartered in Switzerland.
RISC-V International and members
manage and advance the standards.


Alibaba Cloud
ANDES
ESWIN
BOSC
Google
openchip
Phytium 飞腾
QUALCOMM
RIOS
HUAWEI
Infineon
BRAZIL
Microsoft
NVIDIA
SiFive
SYNOPSYS
tenstorrent
ISCAS
ZTE
Tencent 腾讯
10x ENGINEERS
AMD
ANALOG DEVICES
anlogic
MIPS
NEC
NORDIC
inchi
Bluespec
cadence
Canonical
NXP
Rambus
RENESAS
CAST
CDAC
CEVA
CHELPIS
Cortus S.A.S.
RTX
SiFive
SCI
DENSO
E4
SIEMENS
FBMTEK
Fraunhofer
Frontgrade Gaisler
Green Hills
hitex
iar
silansys
SONY
TASKING
igalia
IBM
CETC
THALES
TRUSTONIC
UNISOC
Meta
viettel
XMOS

Shameless Plug: RISC-V Maintainer / Vendor Summit

What: Open Source Maintainers, Distros & RISC-V hardware vendors gathering to discuss issues of the day, longer term plans, call for feedback, etc

When: Last week in January, prior to FOSDEM - 1 day

Where: Brussels Belgium

Cost: Very Low / Free

Contact: tom@riscv.org

Limited sponsorships available

RISC-V Specifications
What's in progress?

How to find things?

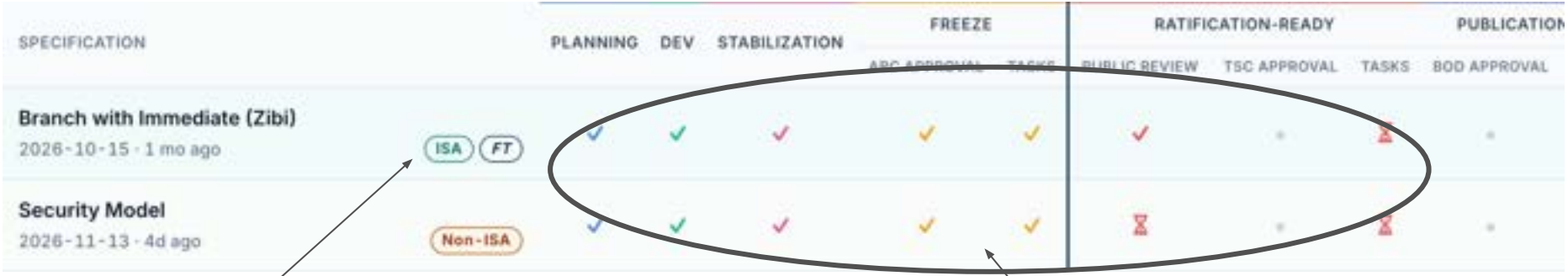
- RISC-V has a **LOT** of ratified specifications, and has quite a number (54!) in development
- Ratified specifications
  - Where? -> https://docs.riscv.org/ (PDF or HTML)
- Draft Specifications
  - **not ratified**, where is the draft? (if available)
    - https://tech.riscv.org/development/
- Getting involved?
  - Join as a member, it's not for everyone.
    - Membership is for technical contributors
  - Task Groups work on specifications
    - Task Groups mailing lists are open (Read only to non-members)
  - Review - Task groups often take material to external communities (GCC/LLVM)
  - Public reviews - 30 days - subscribe to : ISA-DEV
- ISA Explorer (tech preview)
  - https://riscv.github.io/riscv-isa-explorer/
  - Map extensions to instructions or instructions to extension

What's in progress for RISC-V?

- https://tech.riscv.org/development/

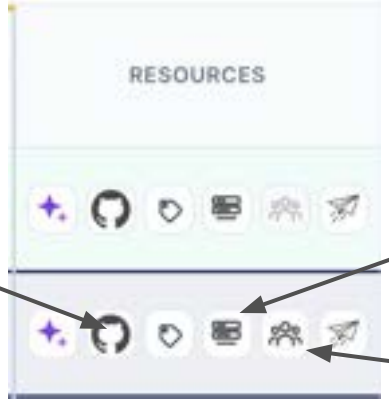


What's in progress for RISC-V?



Type
FT = Fast Track (no Task Group)

Steps completed



Github - ASCII Doc

Current draft (if available)

Committee that is driving specification

Task Group (TG) Information



Mailing List link

Chair and Vice-chair

Task Group Charter

Fast Tracks - no Task Group

ISA Explorer

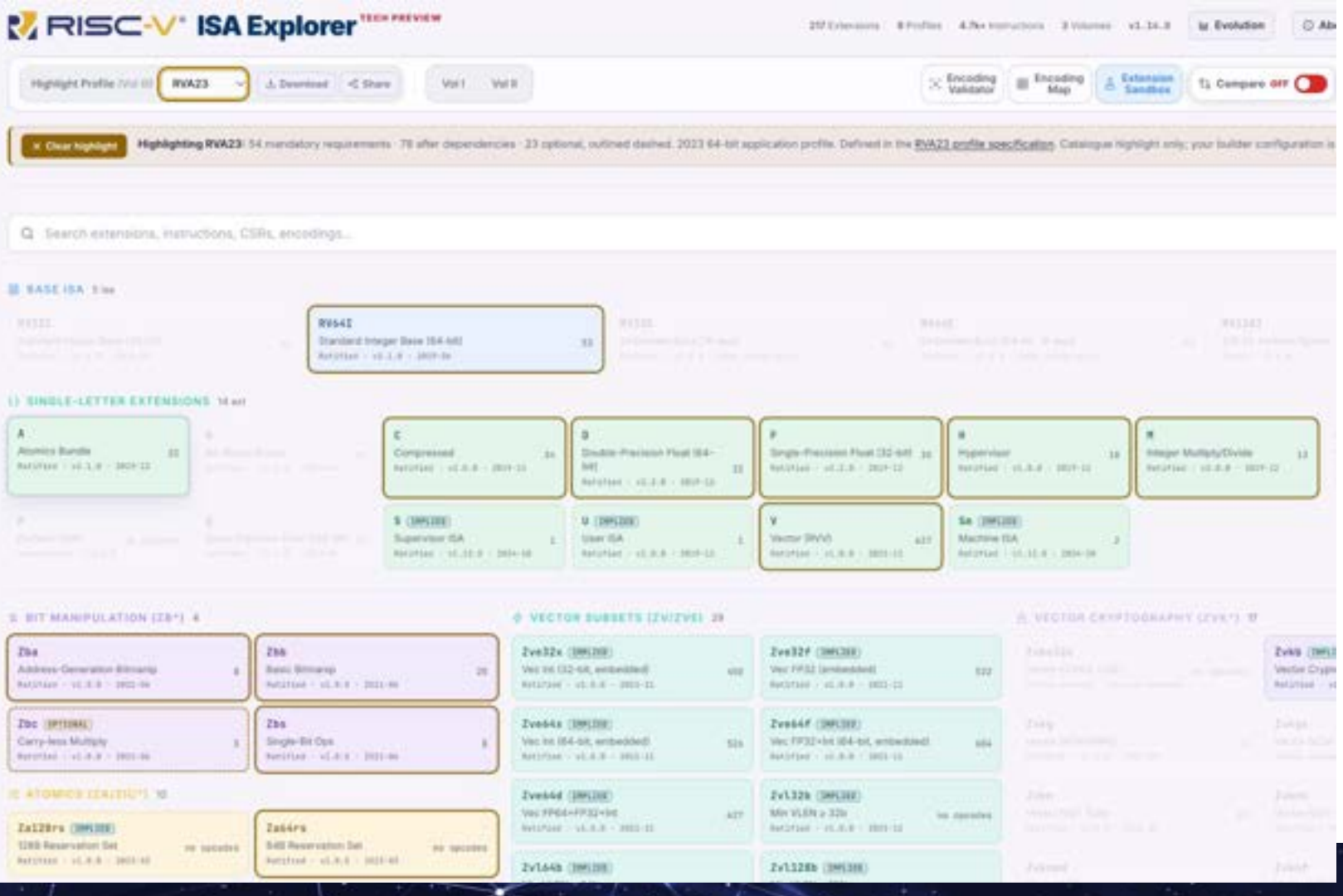
- ISA Explorer (tech preview)
  - https://riscv.github.io/riscv-isa-explorer/
  - Map extensions to instructions or instructions to extension



ISA Explorer

- Explore Extensions
- Explore Instructions
  - Usage
  - Simulator support
  - Toolchain support




RISC-V ISA Explorer TECH PREVIEW
Highlight Profile
RVA23
Download
Share
Vol I
Vol II
Encoding Validator
Encoding Map
Extension Sandbox
Compare OFF
Clear highlight
Highlighting RVA23: 54 mandatory requirements · 78 after dependencies · 23 optional, outlined dashed. 2023 64-bit application profile. Defined in the RVA23 profile specification. Catalogue highlight only; your builder configuration is
Search extensions, instructions, CSRs, encodings…
BASE ISA
RV64I
Standard Integer Base (64-bit)
SINGLE-LETTER EXTENSIONS
A
Atomics Bundle
C
Compressed
D
Double-Precision Float (64-bit)
F
Single-Precision Float (32-bit)
H
Hypervisor
M
Integer Multiply/Divide
S IMPLIED
Supervisor ISA
U IMPLIED
User ISA
V
Vector (RVV)
Sm IMPLIED
Machine ISA
BIT MANIPULATION (ZB*) 4
Zba
Address-Generation Bitmanip
Zbb
Basic Bitmanip
Zbc OPTIONAL
Carry-less Multiply
Zbs
Single-Bit Ops
ATOMICS (ZA*/ZI*) 10
Za128rs IMPLIED
128B Reservation Set
Za64rs
64B Reservation Set
VECTOR SUBSETS (ZVI/ZVE) 29
Zve32x IMPLIED
Vec Int (32-bit, embedded)
Zve32f IMPLIED
Vec FP32 (embedded)
Zve64x IMPLIED
Vec Int (64-bit, embedded)
Zve64f IMPLIED
Vec FP32+Int (64-bit, embedded)
Zve64d IMPLIED
Vec FP64+FP32+Int
Zvl32b IMPLIED
Min VLEN ≥ 32b
Zvl64b IMPLIED
Zvl128b IMPLIED
VECTOR CRYPTOGRAPHY (ZVK*) 17
Zvkb IMPLIED
Vector Crypto
RISC-V


RISC-V Superclusters
Are we there yet?

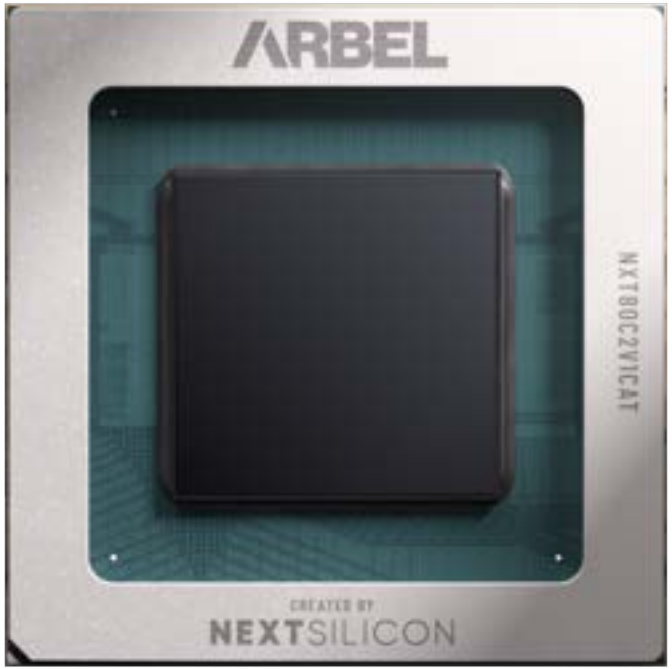
BigSky Datacenter Development Platform

SiFive



- Ready-to-go, fully integrated commercial RISC-V datacenter development platform
- SiFive BigSky P870-D Compute Complex
- 32x P870-D **RVA23** cores operating at 2.0 GHz
- Ubuntu 26.04 LTS
- CUDA demonstration at Hot Chips
  - BigSky connected to an NVIDIA Blackwell GPU
  - CUDA software running on the RISC-V host
  - CUDA kernels executing on the Blackwell
- AMD ROCm™ demonstration at AI Infra Summit
  - Running the Gemma4-E2B LLM model using ROCm 10.0
  - P870-D CPUs acting as the head node
  - AMD Radeon™ AI PRO R9700 GPUs driving inference offload

NextSilicon

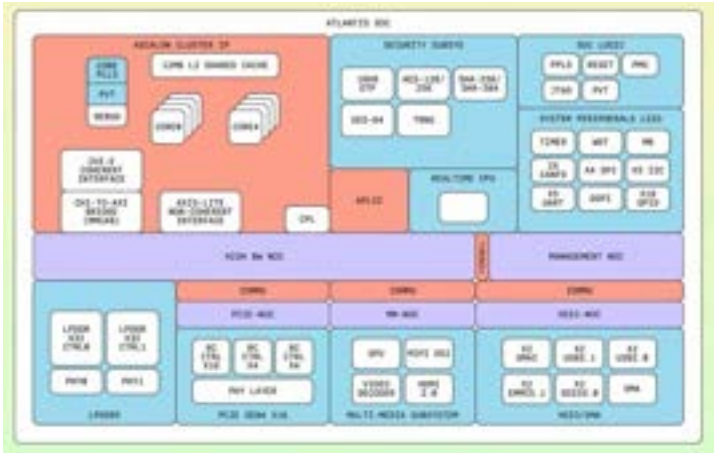


- Maverick-2: The industry's first Intelligent Compute Architecture (ICA) for AI and HPC workloads.
- Delivers performance at scale for the most demanding HPC applications while consuming up to 60% less power than traditional GPUs.
- Arbel: NextSilicon's server-grade RISC-V CPU, designed for next-generation AI and HPC workloads.
- Arbel will be taken from test chip to production, with full **RVA23** compliance.

Atlantis platform

tenstorrent

- Compute platform and development board
- Optimized for HPC, robotics, automotive, and edge AI applications
- 8-core Ascalon X CPU cluster based on the **RVA23** profile
- Atlantis supports the Ubuntu operating system
- Designed for comprehensive ecosystem enablement
- SoC and board bring-up is on track for Q4 2026



RISC-V

SpacemiT K3



- 8 **RVA23** compliant X100 Cores:
  - 4-issue out-of-order execution
- 8 A100 AI Cores:
  - Supports up to 1024-bit RVV1.0
- Up to 60 TOPS of AI performance
- Deployed across applications from robots to servers
- SpacemiT donated 15 boards to RISC-V Developer Program



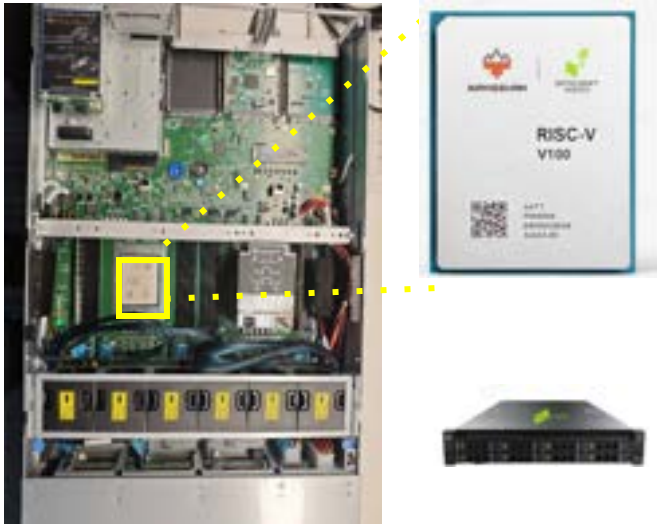








XiangShan-Powered Server Chip



- SpacemiT V100 Server Chip based on XiangShan
- Full compliance with the RISC-V **RVA23** profile, RISC-V Server SoC Specification, and Boot and Runtime Services (BRS) Specification
- 6 XiangShan Kunminghu cores
  - 16.5/GHz on SPEC CPU 2006 Integer
- 40 in-house SpacemiT X100 cores

EPIC Contrail AIX



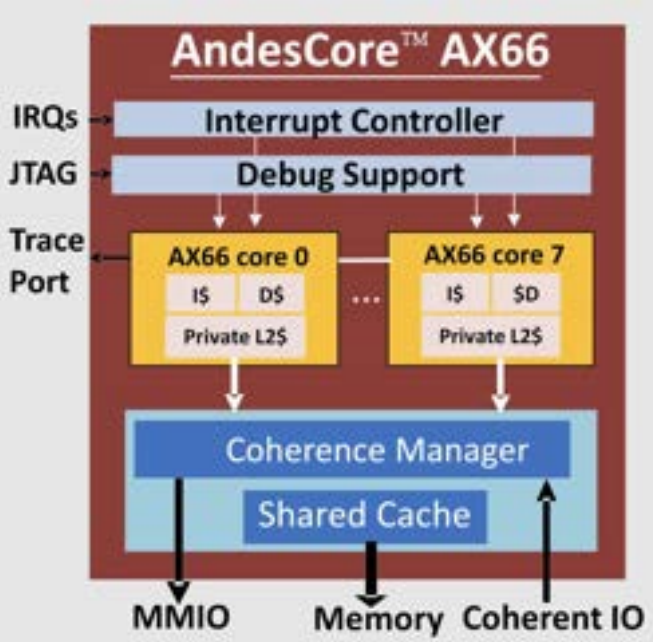
RISC-V Artificial Intelligence eXecution Platform

- Server class RISC-V compute and AI acceleration
- Single unified CPU + AI superchip
- Silicon and server platform available now

Contrail Compute Features

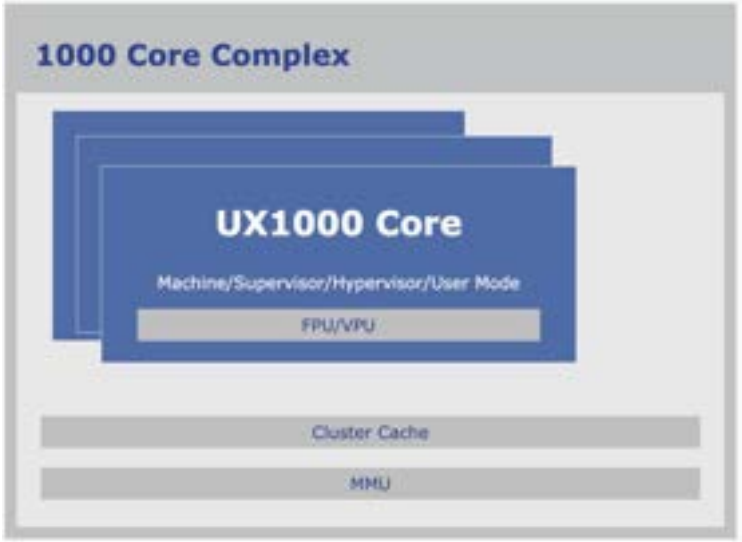
- **RVA23** RISC-V architecture
- 32 RISC-V scalar processor CPU cores
- 16 Integrated AI acceleration RISC-V cores
- Up to 75 TOPS AI processing
- DDR5 memory architecture
- PCIe Gen5
- CXL memory expansion
- Hardware virtualization
- Enterprise-grade security and reliability

AndesCore AX66



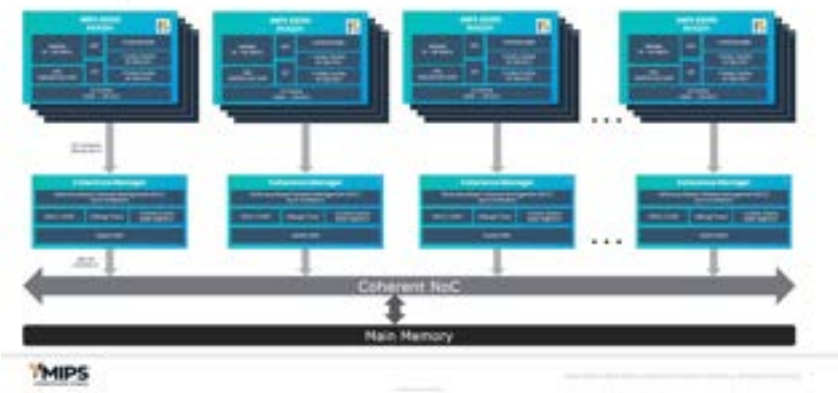
- Andes is ready to welcome RISC-V Silicon for **RVA23**
- 64-bit out-of-order 4 wide decode 13-stage CPU core
- Symmetric multiprocessing up to 8 cores
- Target applications
  - Networking and Communications
  - Android Devices
  - Video and Multimedia Processing
  - SmartNIC or DPU
  - AI SoC
  - Edge Servers

Nuclei UX1030H



- Server, Android, ADAS and robotics applications
- **RVA23** support
- From 2 - 6 issue width
- Up to 8 SMP cores in one Cluster with Cache Coherency.
- Configurable Private L2 cache + Cluster Cache

MIPS S8200



- Software-first RISC-V NPU enables Physical AI at the autonomous edge
- The award-winning MIPS S8200 RISC-V NPU delivers support for modern AI models at the edge, increased efficiency, and performance improvements using an open, scalable architecture
- Enables efficient, scalable Physical AI solutions across multiple markets
- **RVA23**

DC-ROMA RISC-V Mainboard III

DeepComputing



- The world's first **RVA23** laptop
- Powered by SpacemiT K3
- Fully compatible with Framework Laptop 13
- RISC-V performance on a modular, repairable, and upgradeable laptop platform
- SpacemiT K3, 8-core RVA23 CPU
- Ubuntu Desktop 26.04 LTS

Xuantie C950

- **RVA23** Profile
- Up to 8x cores clocked at 3.2 GHz; 22+/GHz Specint2006 base, or a score of around 70 at 3.2 GHz
- Superscalar out-of-order microarchitecture with 8-wide decode
- Floating Point – RISC-V F/D Extension
- Vector – RISC-V Vector Extension v1.0 with Vector Crypto support
- Matrix – XuanTie TPE coprocessor integration



OpenChip - BER10

- Fully functional, Linux-capable 64-bit RISC-V processor built on sub-2nm
- 4 High-performance, out-of-order execution CPU cores
- Wide vector unit purpose-built for AI & scientific math
- PCIe + AXI - Data-center-grade interconnect
- Taped out 2025
- Silicon in hand 2026
- Validation vehicle, available via licensing for partners to test real workloads



2026 is the year of the RVA silicon

Now let’s talk about Software

Back in the day...

Beowulf.org

Beowulf - 1993

1. COTS (Commodity Off The Shelf) based systems to satisfy specific computational requirements
2. Open source software

Beowulf became easy when HOWTO published ~1997, number of installations increased



The Smurf Cluster (1997)

Before IBM Bluegene

Before IBM involvement in Linux

Mix

- Beowulf HOWTO
- Some spare labspace nobody wanted
- Easy access to hardware
- New Token Ring drivers

25 Node Beowulf cluster

Yes Papa Smurf was in charge



Lessons then, lessons now

- Access to hardware
  - Reasonably standard
  - Reasonably performant
- Open software
  - Easily modifiable / adaptable

Creates an environment for results

- Idea exploration
- Inventions - Patent on the right
- New open source software



2026 : RISC-V and Clustering

Personal/Local AI

- Open weight models on local hardware
- Keep data “internal”
- Nothing to the cloud

Other architectures are doing it - What about RISC-V?

On hand....

- Spacemit pico 32 Gig (K3) - Bianbu
- Spacemit pico 16 Gig (K3) - Bianbu
- Deep Computing DC-ROMA III 16 Gig (K3) - Ubuntu
- Qemu - RVA23

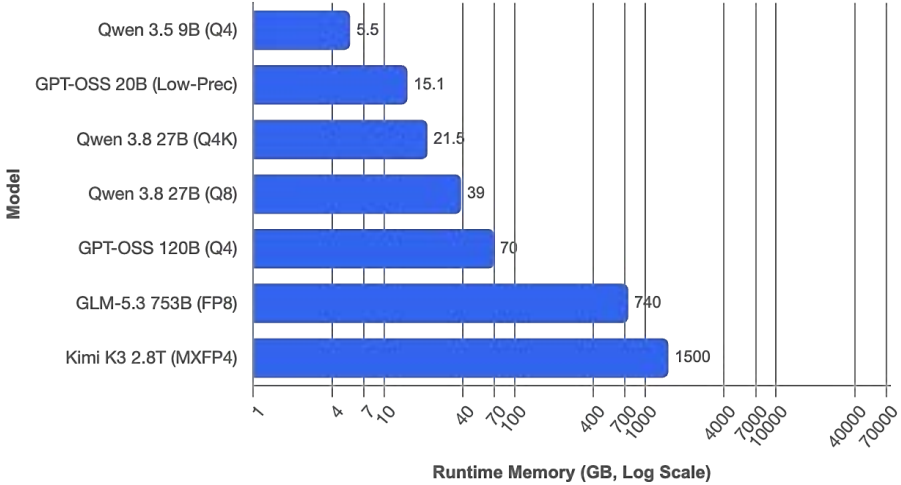


Open Weight Models

Models whose trained parameters (“weights”) are generally distributed under a permissive license and able to be run on a variety of interference engines

RAM usage impacted by:

- Quantized
- Parameters
- Context
- Mixture of Experts (MoE)



It takes a cluster!

Additional Questions for Exploration

What does the cost curve look like for a local cluster of RISC-V hardware?

Is flexibility important for the AI software stack?

Consider: https://runta.com/blog/introducing-frontierharness-eval-v1-1/

- When you switch your model, should you consider switching the harness too?
- What is the best harness for each model?

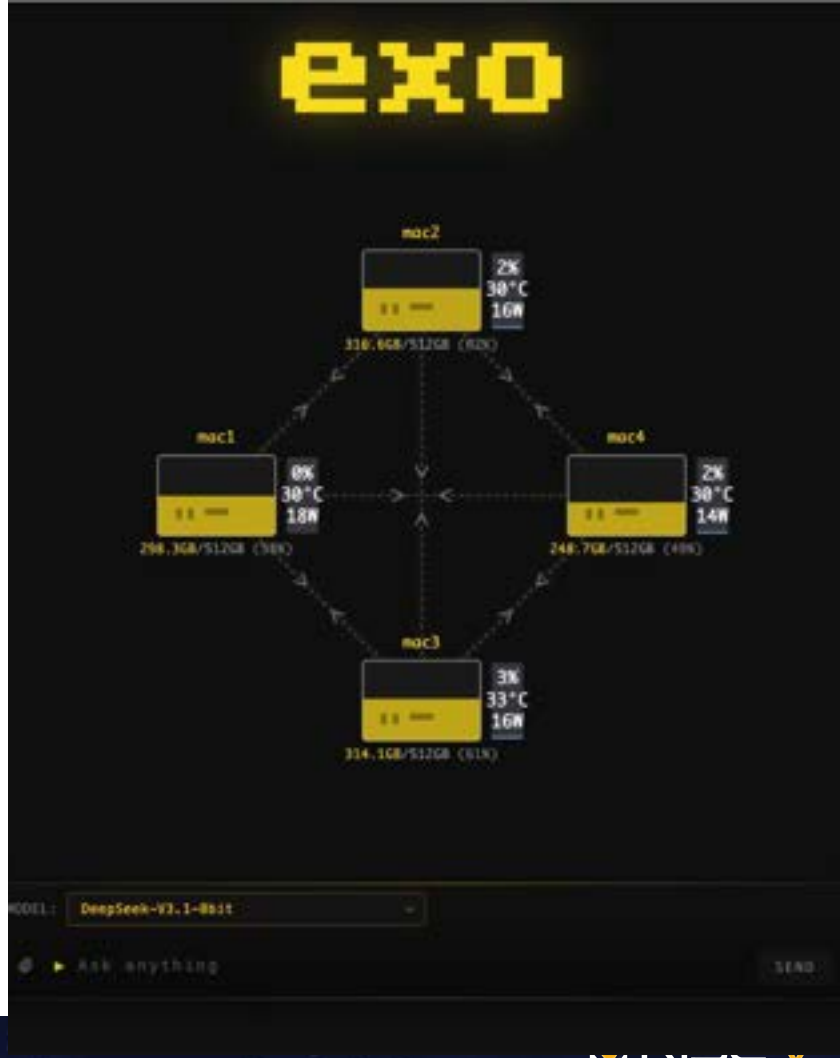


Let’s go!

EXO

- https://github.com/exo-explore/exo
- Open source
- Runs across “diverse” hardware
  - Turns out, not really
  - Macs linked with Thunderbolt 5
- Fairly “easy” to get installed on RISC-V *
  - Running…
  - Buggy

Unmaintained



RISE and PyPI

- AI projects often need various Python dependencies
- Prebuilt binaries are best
  - Source install often includes a build
  - Has package been ported to RISC-V?
- https://pypi.riseproject.dev
  - Hundreds of packages!

pip install --extra-index-url https://pypi.riseproject.dev/



Contribute!

Upstream!

Ray & vLLM



Why Ray?

- Open source
- Infrastructure for distributed execution
- Supports CPU, especially RVV!



Why vLLM?

- Efficient high throughput inference engine
- PagedAttention - Avoid reserving huge blocks of memory to fit a model
- With Ray, scales across machines

Tally Ho!

Ray & vLLM : Challenges





Bazel

- No apt-get install
- **Need:** Packaged native build for RISC-V
- Bazel for RISC-V Howto using Docker
  - https://hartsofthematter.blogspot.com/2026/10/bazel-for-risc-v.html
- 7.5 minimum

Java

- Thanks Spectre for “breaking” Java
  - csrrs rd, cycle, x0 (reading the cycle counter)
  - Impacts java hotspot, which breaks bazel
- sudo sysctl -w kernel.perf_user_access=2

Ray & vLLM : Challenges





Packages

- Py-spy : turn off thread local storage
- Boost : RISC-V ABI breakage
  - raylet.so needs jump_fcontext and friends
  - Boost 1.90 exports a C++ symbols
  - Ray's bundled Boost 1.78 but RISC-V support needs fixing
- Numerous python packages aren't available from PyPI / fail to build - RISE to the rescue! (mostly)
- Arrow, ray-haproxy, OpenBLAS
- PyTorch - vLLM insists on 2.14, available 2.13 is too old
  - Build from source, 2099 files later...

Ray & vLLM : Challenges



- vllm

-- RISC-V detected

CMake Error at cmake/cpu_extension.cmake:264 (message):

RISC-V RVV is available but VLEN could not be auto-detected.

Please specify VLEN explicitly via CMAKE_ARGS:

CMAKE_ARGS='-DVLLM_RVV_VLEN=128' (for VLEN=128 hardware)

CMAKE_ARGS='-DVLLM_RVV_VLEN=256' (for VLEN=256 hardware, e.g. Spacemit X100)



Ray & vLLM : Success?



RAY

vLLM

The journey is not yet complete.

Will we get there?
Yes

CFA: Upstream! Optimize!
Let’s aim for simple installs, optimized experience